

UNCLASSIFIED

AD 419160

DEFENSE DOCUMENTATION CENTER

FOR

SCIENTIFIC AND TECHNICAL INFORMATION

CAMERON STATION, ALEXANDRIA, VIRGINIA



UNCLASSIFIED

NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

**Bureau of Educational Research
Department of Psychology
University of Illinois
Urbana, Illinois**

TEACHING MACHINES AND COMPUTER-BASED SYSTEMS

Lawrence M. Stolurow
Daniel J. Davis

Technical Report No.1
August, 1963

Research on Pre-Programed Self-Instruction And Self-Programed Individualized Education

OFFICE OF NAVAL RESEARCH
Contract Nonr 3985(04)

Principal Investigator
Lawrence M. Stolurow

⑦
TRAINING RESEARCH LABORATORY
University of Illinois
Urbana, Illinois

⑫ **Teaching Machines and Computer-Based Systems,**

⑪ Lawrence M. Stolurow
Daniel Davis,

⑭ **Technical Report, No. 1**

⑪ **August 1963,**

AJ
**Research on Pre-programed Self-Instruction
and Self-Programed Individualized Education**

⑮ **Office of Naval Research
Contract Nonr 3985(04)**

**Principal Investigator
Lawrence M. Stolurow**

TEACHING MACHINES AND COMPUTER-BASED SYSTEMS¹

Lawrence M. Stolurow and Daniel Davis
University of Illinois

The purpose of this paper is to develop a general model of the teaching process as accomplished by an adaptive teaching machine system. In doing this, definitions and distinctions will be made in an effort to provide clarity. No attempt will be made to completely inventory existing equipment or to describe particular machines in great detail. There are many reasons for this, one of which is that an inventory would be incomplete at best. Another is that it would be obsolete before this paper was in print. Furthermore, many of the items it would contain would be unavailable judging from past efforts to provide such an inventory (Stolurow, 1961; Finn and Perrin, 1962). Consequently, it seems most useful now to examine a general model rather than the machines themselves. Hopefully, in doing this the horse will be put before the cart--the teaching machine concept before the machine itself.

Through the development of a proper and complete model of the teaching machine process several advantages could accrue. For example, specific machines could be evaluated in terms of the basic functions they perform and new designs could be developed with a clear perception of the criteria they should meet. Thus, a general and formal model provides both criteria for evaluation and blueprints for design. Furthermore, it focuses attention on a real and important problem long in need of serious scientific study.

¹This paper was made possible by funds provided by the United States Office of Education, Title VII, Educational Media Branch, Contract 2-20-003 and the Office of Naval Research Contract 3985 (04). Portions of this paper were presented at the DAVI-NEA sponsored Invitational Conference on Programmed Instruction, Washington, D. C. on March 24-26, 1963. Reproduction in whole or part is permitted for any purpose of the U. S. Government.

This report is to be published as a chapter in Glaser, R. (Ed.) Teaching Machines and Programmed Learning: Data and Directions.

This is the study of teaching itself. Hopefully, an existing and developing interest in teaching machines will permit us to look at the teaching process with the same sincerity and scientific purpose as in the past, but with a new purchase; namely, freedom to consider the process as accomplished by a device. This new synthetic approach is both challenging and intriguing, for, in general, scientific progress is made when one is freed from subjective thinking. The ability to consider the hitherto personal activity of teaching in a thoroughly impersonal and objective manner should provide a basis for its future development.

Background of Thinking About Teaching Machines

Contrary to popular belief, the development of teaching machines has been a long-term process. As early as 1866, Halcyon Skinner (Mellen, 1936) developed and patented a spelling machine which was conceived as an aid to a teacher, and about 1873 a machine was developed with generated solutions to logical problems which were presented symbolically (Jevons, 1958). In 1915, a teaching machine of the type that is so common today was developed (Pressey, 1926). It is interesting that none of these early devices for automating particular teaching functions struck a very responsive cord.²

The original Pressey teaching machine and that of Ordahl and Ordahl (1915) were neither simpler nor less glamorous than the later devices of Skinner which did spark interest. All did just three things: displayed

²Somewhat earlier, Thorndike (1912) had suggested the idea of a response-dependent display device, but the reinforcement function per se was not made an explicit factor. However, in this statement, Thorndike did specify "...only to him who had done what was directed on page one would page two become visible, and so on,...". Consequently, reinforcement was implied through the different consequences attendant on right and wrong responses. In retrospect, Thorndike's conception reads like a blueprint of today's printed devices, yet it too failed to ignite either the interest of educators or the imagination of researchers.

stimulus materials, accommodated a response and provided reinforcement. The last of these three functions, reinforcement, was considered the most important and critical for a teaching machine by both Pressey and Skinner.

Pressey had set for himself the task of designing a machine that both "tested and taught." In order to test, the machine had to display materials and accommodate a student's response; to teach, it also had to tell the student whether or not he made a correct response. Thus, the feedback in terms of automatic reinforcement or knowledge of results took on a special definitive significance. It was seen as critical for a machine that taught. While memory drums did this, they did not do it automatically. A teaching machine needs to provide the student automatically with reinforcement and to do this, it uses the student's own response.³

Since the early Pressey and English teaching machines, a variety of other simple devices designed to deliver automatic reinforcement have been developed, tried out, and found to be effective (Pressey, 1950). Thus, Pressey first demonstrated, through research, an effective and deliberate transplant from the learning laboratory to the classroom. Furthermore, in doing this, he used an automatic mechanism.

During World War II (See Valentine, 1947; Vallance and Shraeder, 1947) an extensive set of devices was developed to teach skills by individualized self-instructional methods without the intermediation of a teacher. Each step of a skill, such as the disassembly-assembly of a piece of equipment

³ H. B. English (1942) invented a device used in 1918 to help train soldiers squeeze a rifle trigger which provided visual feedback through the use of a manometer revealing to the soldier a change in the height of a liquid column that rose slowly if he squeezed or spasmodically if he didn't.

(e.g., a machine gun or turret) was programed and an overt, constructed response required. The student's task was to accomplish the steps and learn the skill. After practicing until he was proficient the "phase check" was administered by an instructor. This means that the student performed the same steps, but now he did them without the program. The instructor used it as his record sheet to identify the student's errors. In effect, the phase check procedure also provided a vanishing condition since the verbal cues were removed and the trainee had to perform the steps without them. In fact, vanishing was carried even one step farther when the trainee was blindfolded and required to perform the skill (e.g., disassemble a machine gun) without seeing the parts. Here, feedback was automatically provided by the accomplishment of each step with the actual equipment. The student either removed the part (e.g., the back plate) or he didn't. There was no need for externally provided knowledge of results. Feedback was provided for each step and every step had to be performed correctly before the next one could be done. Students progressed at their own rate and were scored on the phase check in terms of both errors and time. There was no external test since the objective of the program was to teach the very skill that was practiced and on which the trainee was checked.

Thus, we see that the basic concepts of automatic feedback, providing knowledge of results, and the use of sequentially ordered steps anticipated the subsequent labeling of the device and explication of the conception by Skinner (1954).

Some thirty years after Pressey's first published description of his teaching machine (Pressey, 1926), when there was no depression, surplus of teachers, or war, interest in the possibilities and potentialities of automated instruction was sparked by Skinner (1954; 1958). The idea of

automating the classroom was long overdue relative to other areas. In fact, instruction seemed to be the last frontier for the application of cybernetic notions (Weiner, 1948). It was well established that learning required feedback and a simple extension of this clearly revealed the necessity and advantage of controlling the learning by mechanisms which automatically provided feedback to the student.

In reviving and redirecting thinking about automated instruction, Skinner, like Pressey before him, focused attention on the "teaching machine," the hardware; but unlike Pressey, he gave greater prominence to the "program." In fact, Skinner emphasized the significance of the stimulus materials, and used them rather than the machine to provide the feedback. He also related his thinking about operant conditioning to programmed instruction through the notion of "shaping" which can be accomplished only if the material is organized. The organization of the verbal material (the steps of a program) becomes especially important when you think of it, as Skinner does, in terms of the program and machine doing all the teaching. It is this departure from Pressey's conception of the machine as an adjunctive device used in combination with other instruction that raised questions of strategy which are so prominent today. For Skinner the machine was an instrument that taught independently of other means of instruction; it was not merely a testing device.

The Current Conception of Teaching Machines

The current conception of the teaching machine and of its value relative to that of the program is an obvious switch from the perception of its relative value by Pressey and Skinner. There has been a rapid turn-about resulting in the dismissal of the machine as an important instrument in programmed

learning. Instead of the machine being the great hope of the future it now is either tolerated or thought unnecessary and programed instruction is seen as achieving its effectiveness through the functions performed by the "program." The software in automated instruction now is perceived as being more important than the hardware (Goldstein and Gotkin, 1963, Stake, 1963).

This view seems to have resulted from the separation of the functions now associated with the software (paper or film) from those associated with the hardware. The simple nature of most teaching machines currently available allows both the content and logic (decision functions) to be kept together on paper or film. Therefore, the paper or film (the program) becomes the essential item while the device in which it is placed for display (the machine) is merely auxiliary. The position taken here is that this simple division of functions is ill-advised and inadequate. Although it describes most simple teaching machines, it cannot be extended to more adaptive and complex machines where the content and logic are physically separated. It will no longer be possible to separate "machine" functions from "program" functions according to physical location. They both will be accomplished by a complex combination of display, switching and computer operations.

Toward a General Conception of Teaching Machines

First, it should be realized that the concept of the teaching machine is not uniquely defined by any existing machine, and second, that existing machines give a limited and biased view of the concept. Therefore, when asked about what a machine is, we should not think of teaching functions that are now accomplished by a machine, but rather of functions that can be accomplished by a machine. Thinking in the former way, as many do,

indicates that our current concept of a teaching machine is poorly defined and in need of reformulation.

In most general terms, a teaching machine is a response dependent, adaptive mechanism which produces systematic behavioral changes. Here "mechanism" is used in the abstract sense referring to a set of specific functions or transformations. Teaching machines we can touch are exemplars of the general concept, and implement specific functions associated with teaching. No actual machine should be confused with the teaching machine concept itself.

The specific way in which a machine is made to work is a separate question from that of the functions it performs. The former involves synthetic operations while the latter is analytic. Therefore, when we consider teaching functions for the purpose of machine design, there are two fundamental problems. One is the set to be considered and the other is the specific manner in which the functions are to be accomplished. Whereas the former is determined by an analysis of learning and the objectives of instruction, the latter is decided by such practical factors as cost, convenience, state of the art, etc.

Toward a General Conception of Teaching Programs

As defined above, a teaching machine makes use of response-dependent information to bring about systematic behavioral changes (usually changes in performance level). This transformation is accomplished by means of the teaching program which is made up of two interacting parts. The first, the content set, consists of expository and interrogative materials divided into small units (frames). The second, the strategy is a set of decision

rules which is based on the interrogative content and which controls the presentation of content units.

It is important to notice that this definition of a teaching program says nothing about software or hardware. The physical location of either the content set of the strategy can vary from paper or film in the simple machines to magnetic tape or core storage in more complex systems.

On the basis of these definitions, it becomes clear that any comparison of machines and programs is meaningless. They are not comparable. Anything which uses response-dependent information to change performance levels is a teaching machine (whether it is a book or a computer-based system). Also, every teaching machine, no matter how simple or complex, uses some sort of teaching program to bring about the desired change.

The Two Phases of Teaching Machine Operation

In order for a teaching machine to be maximally adaptive, it should have the following capabilities. First, for a given subject area it should have at its disposal a number of different teaching programs. It is clear that no single strategy will work for all students, and that no single type of content is best for all. Since a program is made up of a strategy and a set of content units, it follows that no single program will work for all students.

Second, when specifying a program for each student, the teaching machine should make use of past performance and ability measures. This information is necessary if it is desired to use the "best" teaching program for each student.

Third, the teaching machine should have the capability of changing programs during the course of instruction. Getting the "best" program for each student is not always possible. Therefore, if it is found that a program is not effective, then a new one may overcome the difficulties which the old one could not handle.

These capabilities are imbedded in the model of the teaching process which is developed in the following sections. The model divides the teaching process into two phases. The first, the pre-tutorial phase, specifies a teaching program which can attain a desired outcome subject to the ability constraints of the student. The second, the tutorial phase, implements and monitors the program which was specified in the pre-tutorial phase. The monitoring process allows the teaching machine to change programs if the student exceeds certain limits (both upper and lower) in level of performance.

The Pre-tutorial Phase of Instruction

Variables

The pre-tutorial process can be described in terms of three basic sets of variables. The first set is made up of the possible outcomes of the teaching, and it contains a subset which consists of the desired outcomes or objectives. It is necessary to have the objectives clearly in mind when specifying a teaching program for each student.

The second set is made up of the possible "entry behaviors" of the students, and the third contains the teaching programs which are available. Thus, the pre-tutorial process deals with relations between outcomes, entry behaviors, and teaching programs. The goal is to specify a teaching program which will accomplish a stated objective for a given entry behavior.

Characteristics of the outcomes. A closer look at what is meant by an outcome reveals three characteristics. It is a level of performance for a given subject area which is attained within a given time. Therefore, when one describes the desired outcomes or objectives of teaching it is necessary to specify three things: (a) the subject area or topic; (b) the final level of student performance that would be the minimum acceptable; and (c) the maximum time allowed to achieve the minimum level. A proper statement of objectives is readily translatable into training specifications, on the one hand, and into the requirements for performance scores of the student, on the other.

It is assumed that the achievement of an objective for a particular student requires a teaching program. Consequently, if any of the three characteristics (topic, level, or time) is changed then the objective is changed and a different program is required to achieve the new objective for the student with a particular level of entry behavior. For example, if the minimum acceptable level of terminal performance is raised, but the topic and time are held constant, then a new program is required to get a student who has a specific level of entry behavior to meet the objective. For example, it might be necessary to give more opportunity for practice. Similarly, if the time available for teaching is reduced but the topic and minimum acceptable level of final performance are not changed, a new program is necessary for students of a given entry level (e.g., the concepts might be organized differently).

Frequently, it is necessary to change deliberately a second characteristic of the objective when one characteristic has been changed (e.g., by external factors). For example, if the amount of time available is reduced, then it

may be the case that no known program can be found to achieve the final level of performance with the given students. If this is the case, then it may be necessary to alter either the topic covered or the time allowed, or both. By reducing the scope of the topic or by increasing the time, the new objective may be achievable with the given students using an existing program. However, if this is not the case, then it may be decided to change both of the other two characteristics so as to make possible the achievement of another objective. The implication is clear. If one characteristic of the objective is changed, a change in another is also required when the same program is to be used with students of a given entry level.

In many cases where the word "objective" is used, it has a more limited meaning than the term is given here. For example, many teachers define their objectives simply in terms of the minimum level of performance they will accept. Consequently, they may say they get all their students to meet the same objective when they achieve a satisfactory score on a test. From the present point of view they do not do this. To indicate more specifically why this is so, consider the case where all students taught by a particular program achieve a minimum score on the achievement test used. Among these students, the different ability levels will have used different amounts of time to complete the material. Also consider their terminal level of performance assuming that they all achieved the minimum test score. It, nevertheless, will be found that there are distinguishable final test score levels. If we define the objective in terms of time required to achieve a particular level of proficiency in a topic, then these differently performing students are really achieving different objectives. To consider only the final test score is to court confusion in thinking about the problem.

Characteristics of entry behaviors. There are two critical characteristics of the entry behavior (B_e) of the students. The first is their level of performance on the immediately preceeding relevant task or on a pre-test (P_e); and (b) their aptitude level (A_p). Usually it is possible to specify these characteristics reliably and with acceptable precision. In order to determine the student's level of entry behavior, it is necessary to give him tests, however, and these tests are accomplished before he takes the instructional program. The scores obtained can be used for selection of the students for training. Usually, the level of entry behavior (B_e) varies among the students selected, but for any one student it is a fixed value at a particular time. Nevertheless, even for the individual student, it is potentially variable over time. Consequently, it is possible to alter a student's entry behavior before he takes a program. For example, this could be done by giving him review sequences or additional practice on the task just completed so as to increase his level of performance on the pre-test (P_e). It is more difficult to change the student's aptitude level, particularly as he gets older and as the available time is reduced. Aptitude is the more stable component of entry behavior.

In experimental studies, the students' entry behavior is typically an independent variable and students are grouped in terms of selected levels. Then, one treatment is given them and differences in final performance observed. In the usual training situation, on the other hand, some screening procedure is used to select students. Its purpose is to minimize failure or to maximize successful training in the allotted time with the training strategy constant. Therefore, in both research and in practice there is a recognition of individual differences in response to a treatment or procedure.

When different levels of entry behavior are known to exist among a group of students, then choice is possible between the use of the same objectives for all of them or of different objectives for various entry levels. In the practical situation, there is associated with each of these alternatives a different requirement for the number of teaching programs if it is desired that all students within a given range achieve the same objective.

Characteristics of teaching programs. The programs are the means used to transform the student's entry performance level (P_e) into a specific final level (P_f). Unfortunately, the state of the art is such that it is not uncommon for a program to be developed independently of a specification of entry behavior or final performance. Then it is tested to find the objectives which can be accomplished with it as revealed by the time it takes a selected group of students to make a satisfactory score on a test. This situation is unfortunate in the sense of not being ideal; however, it is not unusual nor likely to be permanent. In fact, it is characteristic of the immature state of programing as a technology. Not too long ago, when the technology for producing transistors was at a similar state of its development, the procedure used was akin to that described for the development of programs today. A large batch of transistors was produced and then each was submitted to tests to determine its characteristics. Once characteristics were specified, the transistor was typed or classified. This ad hoc procedure was necessitated by lack of specification of the production procedures (programs) in relation to particular input-output relationships. In the design of an instructional program, we do not now have adequate data to permit us to specify one for a specified change in entry behavior to achieve selected objectives.

As was mentioned in a previous section, a program is made up of two interacting parts: the set of content units and the set of decision rules (strategy) which are defined on the content units. It is possible to change a program for a given topic by changing the type of content that is used (e.g., from a geometrical to an analytic presentation in mathematics). Also, it is possible to change a program by specifying a new set of decision rules (e.g., change the number of responses used in making decisions or change the amount of remediation given for errors). Thus, a complex teaching machine would have at its disposal a number of different types of strategies which, when combined with different sets of content units, yield a wide variety of programs.

Pre-tutorial Decision Process

The Pre-tutorial Decision Process, which has as its goal the selection of a suitable program, is described as follows:

Step 1. Search for a Program. In order to identify a teaching program, it is necessary to know two things about the student and three things about the objectives. For every student, it is necessary to know: (a) his aptitude level-- A_p ; and (b) his performance level on the immediately preceding relevant task or on a test of what is to be taught-- P_e (entry performance). For the objectives, it is necessary to know: (a) the topic-- T ; (b) the minimum acceptable final level of performance to be achieved-- P_f ; and (c) the time within which the student must achieve the final level-- t . With this information as input, then the search for a program begins. The outcome of the search may be any one of three types:

a. More than one may work. If the search reveals that there is more than one available program which will transform the entry level to the

desired final level, then these must be listed so they can be evaluated in terms of efficiency criteria.

b. None may work. If no available program works, then one or more of the objectives will have to be changed.

c. One may work. Only one of the available programs may be found to work, and if so it is identified.

Step 2. Evaluation. This step varies depending on the outcome of Step 1 as indicated.

a. When more than one program will work, then there is a problem of deciding which to use. There is no unique answer to this type of problem. Time and cost are usually critical parameters. The program that takes least time or costs least is the one used. However, educational rather than economic factors also may be used. For example, the "best" program may be the one that can be used over the widest range of talent. For this outcome, it is necessary to have an explicit set of criteria to use in deciding among alternatives. Only one program can be used at a time.

b. When no available program will work, the problem is to decide what to change--inputs or objectives. The priority order in which the changes will be considered will depend upon the particular situation, however, a general pattern can be described. The basic decision is between changing the students or the objectives. For example, if student changes are considered the minimum entry level (P_e) may be reduced by a certain amount (e.g., to $P_e - x$) for all students within a particular aptitude range (e.g., $l < A_p < h$). On the other hand, a decision might be made to improve students who fall within a specified range of the minimum entry performance level P_e . In other words, for all students whose entry performance (P_e) is less than P_x but above P_y , review exercises might be given so as to bring them to the minimum entry

level of performance for starting the program. If, on the other hand, the objectives are changed, the most likely change is in the time allotted to the student. The second is the amount of content or definition of the subject area; the least likely characteristic to be changed is the minimum acceptable final performance level.

c. When only one program works, it is identified.

Step 3. Implementation. Once a program is specified it can be put to work.

The decision processes just described can be represented as a flow diagram and can serve to describe one basic set of things that a sophisticated teaching machine might be designed to do. This is illustrated in Figure 1.

Relating this process to a machine suggests that the input to the machine would be information about the entry behavior of the students in the terms already indicated. Given this information and data on objectives to be achieved, the machine would carry out definite procedures to make appropriate decisions about the program to be used. Figure 1 represents a general model of the teaching process as accomplished by an adaptive teaching machine. It contains as its last step the implementation of a program as the tutorial decision process. This is discussed in detail in a subsequent section.

The Pre-tutorial Process as a Two-Person Game

It is possible to consider the set of relationships in teaching as a process involving a single student and a teacher in interaction. In doing this,

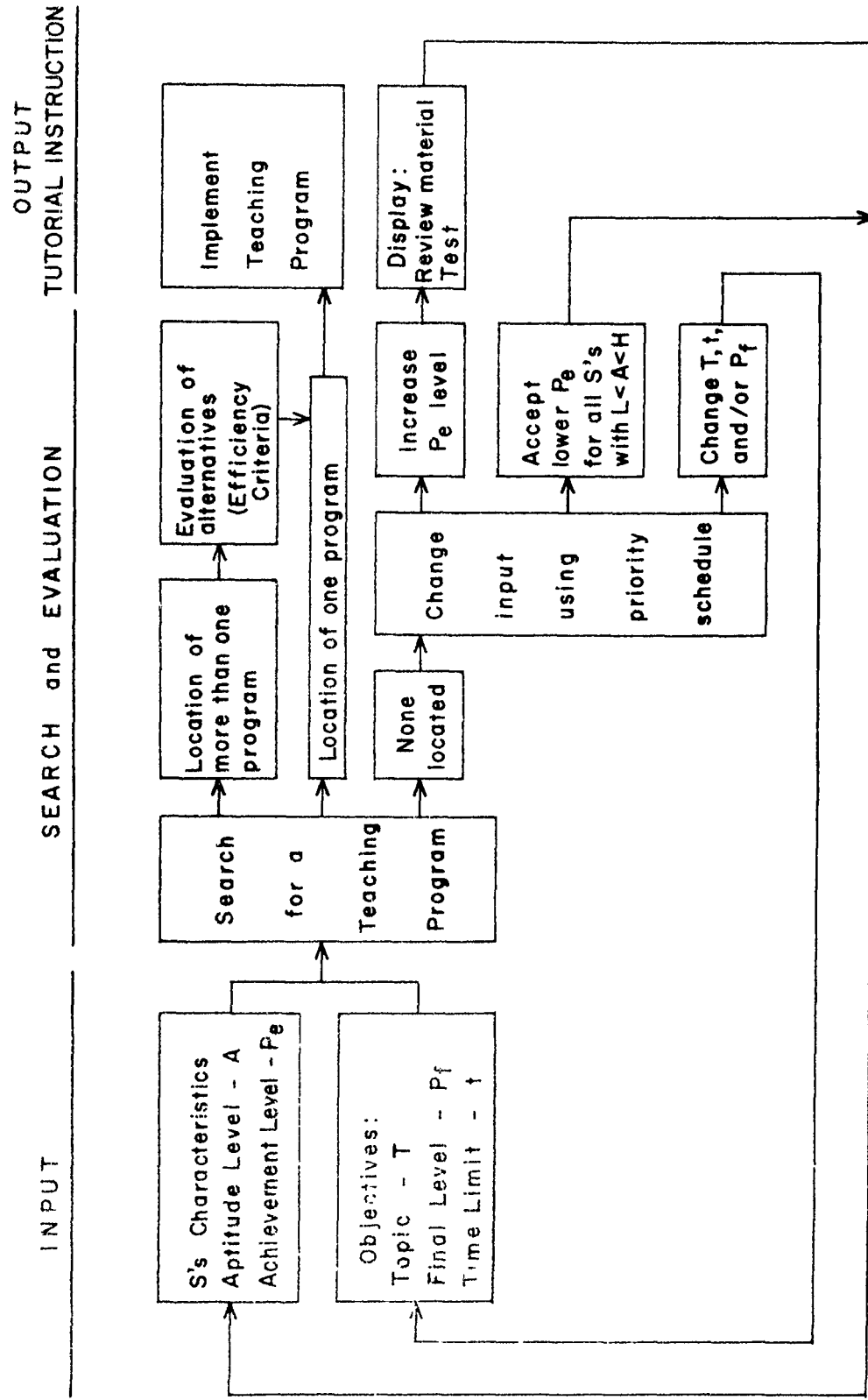


Figure 1. The Pre-tutorial Decision Process

the process is represented as a matrix using the sets of variables already described. This is done in Figure 2 with three elements per set--teacher's programs (α, β, γ), student's entry levels (a,b,c), and possible outcomes (1, 2, 3).

In using this form of representation, for example, we can select a particular objective (e.g., number 1) for a student (S) whose entry behavior is represented by a in the matrix. Having done this, the teaching strategy is determined and is β . If, however, S's status were b, rather than a, then the required teaching strategy to achieve objective 1 would be 2 rather than β .

It is apparent that it is necessary to know three things in order to set up a matrix representing instruction. The outcomes need to be specified, the teaching programs must be known and the entry behavior of the students must be determined. Any one of the outcomes can be achieved by applying program α, β, γ to students with a known level of entry behavior. With the example given in Figure 2, the teacher's position is ideal for he can always achieve a stated objective by simply using one of the available programs. T (the teacher), in fact, has complete control over the outcome, for every row has at least one of the possible objectives in it.

Consider another example. Figure 3 presents a different matrix of the same general form. With this matrix things are different, however. The difference is not just in size, as an examination in greater detail will show. For example, if S's status at the beginning of the program is c, then T has several ways of achieving outcome 1--by using program β, γ or δ . If,

		Teacher's Programs		
Students		<u>S</u>	<u>F</u>	<u>O</u>
<u>S's</u>	<u>a</u>	2	1	3
Entry	<u>b</u>	1	3	2
Level	<u>C</u>	3	2	1

Figure 2. A 3 X 3 matrix representing instruction.

however, outcome 2 were the objective instead of 1, then T could not always achieve his objective, for if S's entry status was b or c none of the programs available to T would work. Students with entry status b, for example, can only be made to accomplish objectives 1 and 4. The same is true for those whose entry status is c, but for those whose entry status is a, d or e objectives 1, 2 or 4 can be achieved. This approach recognizes that it is not always possible to achieve every objective with a given set of programs.

It is apparent that different arrangements within the matrix and different numbers of programs and states available to T and S, respectively, can give rise to a variety of situations with respect to what T can do. In order to copy with the many possibilities that can arise, it is necessary to set some restrictions on the population of matrices with which we can deal, and in doing this we want to maintain sufficient complexity to be of interest. One type of matrix that does allow a good deal of variability and some measure of precision is the set in which no column contains a repeated outcome. Whenever this is the case, T must select a strategy taking into account S's entry behavior in order to accomplish the stated objective. Every S requires a different program if a particular objective is to be accomplished. The relations between columns and rows are unrestricted in all respects so that for any r x c matrix there are a number of possible combinations leading to different outcomes. Under these conditions the teacher, or teaching machine, needs to be discriminating in specifying a program. Figure 4 is a 3 x 9 matrix of this type. Once an outcome is specified, T must decide what program will be used for each type of entry behavior (a through i).

		T's Programs			
		<u>a</u>	<u>b</u>	<u>c</u>	<u>d</u>
<u>S</u> 's Entry Level	<u>a</u>	2	4	1	1
	<u>b</u>	1	4	1	4
	<u>c</u>	4	1	1	1
	<u>d</u>	4	2	1	2
	<u>e</u>	4	1	2	4

Figure 3. A 4 X 5 matrix representing instruction.

One and only one teaching program (γ or δ) is possible in order to achieve a particular objective. For example, if S 's entry status is a , then to achieve objective 11, it is necessary for T to use program γ . When S 's entry status is b then T has to use program δ , and when it is c he must use program β .

Implications of This Analysis of Teaching

The program as a transformation uniquely specifies an outcome for a student and a set of outcomes for students who differ in entry behavior. If, for example, only one of the programs (Figure 4) were used with a group of S s then every S would achieve a different score. The outcomes (numbers) in the matrix might be levels of performance on a valid achievement test divided by the time the program takes.

If only one outcome is accepted as the objective, and the students differ in their entry behavior, then the teacher, or machine, must use a different program for every level of entry behavior. If more than one outcome is accepted, then many different programs still are required, but not as many are needed as when only one outcome is sought. In the example given in Figure 4, the best T can do is to use one of three programs (e.g., once for each of the first three S s) and then he must repeat one of the programs for the fourth S . Obviously it is impossible to get all S s to earn the same score unless T has available to him as many different programs as he has types of students in terms of their entry status. This useful implication can be summarized in this way. If no two outcomes in the same column are equal, and if a set of outcomes is selected by T , one from each

		<u>T's Programs</u>		
		<u>α</u>	<u>β</u>	<u>γ</u>
<u>Ss' Entry</u> Levels	<u>a</u>	6	6	11
	<u>b</u>	11	5	6
	<u>c</u>	13	11	1
	<u>d</u>	2	2	2
	<u>e</u>	3	17	3
	<u>f</u>	8	8	13
	<u>g</u>	10	4	4
	<u>h</u>	1	16	9
	<u>i</u>	12	14	8

Figure 4. A 3 X 9 matrix representing instruction.

row, and if the table has r rows and c columns, then the number of different outcomes cannot be fewer than r/c .⁴ In the case of Figure 4, the minimum number of outcomes is $\frac{9}{3}$ or 3.

Some Implications for Teaching Machine Design

One thing this analysis implies is that every machine that is limited to a single strategy and set of content units will produce as many outcomes as there are different types of entry status since r/c would reduce to r , with c equal to 1.

This implication may seem unacceptable, for it might be asserted that a machine which accommodates only a linear program could be used with a variety of different Ss in terms of measured entry behavior and yet all of the Ss might answer every question on the test given when they completed the program. While this might sound like an empirical denial of the analysis, it is not. There are many things that are unspecified in the description which could make the example not fit the model. To get a purchase on the problem, let us re-examine each class of events in relation to it. Since the program is specified, there is no ambiguity there. The other two classes of events, however, could present problems in fitting this model to the hypothetical data. If we accept the achievement test as both a valid and sufficient index of the outcome, then the distinctions among Ss was invalid. All Ss must have actually been alike at the beginning, for a program transforms a given entry behavior into an outcome. This says that different values should appear in the matrix for each combination of entry level and program. However, all scores were said to be the same, e.g., 100%.

⁴This is based on the "Law of Requisite Variety" as presented by Ashby (1961, Chapter 11).

With the hypothetical situation just described, the conclusion is that the entry behavior measures are either irrelevant or invalid. The model, as contrasted with the example, simply says that we cannot "eat our cake and have it too." As long as we accept the notion of individual difference in entry behavior it follows that the outcome of a single teaching program will be the achievement of different outcomes. In other words, different values (representing outcomes) must appear in the rows of any one column if the entry behavior measure is valid and the test used at the end of instruction is a valid measure of the outcome.

All students may achieve the same score on a valid test of objectives but they will not do it in the same time, for example. However, Ss, at different entry behavior levels may achieve the same score in the same time if more than one program is used.

If we do not find the number of different outcomes equal, at a minimum, to the number of valid classes of entry behavior something is wrong in one of the following ways. The classes of individual differences are invalid; the test of attainment of objectives is insensitive and does not provide the degree of discrimination that is required; or the programs are not really different.

It is useful to distinguish between the objectives set for a group of students when the content on which they are working is held constant from the objectives set for a single student when content varies. The former are group objectives and the latter individual objectives. In teaching, the generally accepted goal is to minimize the number of outcomes which are accepted as group objectives. If this goal is used, then it follows that a large number of teaching programs is required if the entry behaviors vary

over a broad range. Concerning individual objectives, it is generally desired to maximize the number of objectives which each student can achieve. Accepting this as a goal will also require a greater number of teaching programs when the number of entry levels is increased.

The implications of this analysis for the design of a teaching machine are clear. A machine must have the capability to handle program decisions if it is to be used to teach students who vary widely in entry behavior. In fact, it goes further and says that the broader the range of entry behavior to be taught the larger the requirement for using different programs if we want all students to achieve the same objective. This means one of two things. The first is that a battery of simple machines and linear programs can be used, one with each entry behavior level. The second is that a computer-based teaching system can be used to cope with a wide range of individual differences in entry behavior.

Since the first alternative means that a great deal of program development would be required and this is not easy to accomplish, the computer-based system seems to be the more promising alternative. Here, different strategies and sets of content units can be combined to generate a large variety of programs. To reduce the requirements placed upon the computer, it is necessary to standardize input and output.

The Tutorial Phase of Instruction

Thus far we have talked about the decision process which is required before the tutorial process is begun. Once a strategy and set of content units are selected for a student, the program is implemented in the tutorial phase of instruction.

The tutorial process can be defined in terms of three sets of variables. The first set is made up of response measures such as latencies or number of incorrect responses for a given number of content units. The second set consists of the content units, and the third set consists of the decision rules which relate the response measures to the presentation of subsequent content units. For a given program (set of decision rules and content units), the values of the response measures obtained from the student determine the instructional sequence. This can be illustrated by considering a branching program of the Crowder type (Crowder; 1960). Here, one response measure is used which can take on a number of values: correct response; incorrect response-A; incorrect response-B; incorrect response-C; etc. The content units are divided into two sets: the main sequence units and the remedial units. Each main unit has associated with it a number of remedial units. The decision rule is as follows: for each main unit look at the student's response; if it is correct, present the next main content unit; if it is incorrect-A, present remedial content unit-A; if it is incorrect-B present remedial content unit-B; etc. Thus, the decision rule specifies the next frame on the basis of the student's response.

In computer-based systems the decision rule will, in general, make use of more comprehensive response information (e.g., latencies, weighted averages

of past responses), and will be able to specify a greater variety of instructional sequences on the basis of this.

The interaction of the program and the student as described here is much like that of a teacher and a student. Therefore, to distinguish the implementation of a program from the monitoring of a program (which is discussed shortly) it will be referred to as the Teacher Function. The Teacher Function, then, involves the use of a decision rule to choose subsequent content units on the basis of response information.

It is also possible to build into a teaching system the capability of learning about the student as he responds (Pask, 1960). On the basis of a monitoring process, changes in the program itself can be made (e.g., either content or strategy) if the student's performance exceeds certain bounds. If it is found that the strategy being used is too demanding, then one which is more suited to the student can be used (with the content fixed). On the other hand, it may be found that the type of content being used is not suited to the student, in which case a new set is specified (with the strategy fixed).

The monitoring of a program will be referred to as the Professor Function. This involves dynamic decisions which specify changes in the program on the basis of student performance. The performance measures which are used on this case are more comprehensive than those used in the Teacher Function. In fact, ability measures may be used as well in a decision process similar to the one used in the pre-tutorial phase (Figure 1).

A diagram of the interaction of the student, teacher, and the professor is illustrated in Figure 5. Here we see two-way communication from the student

to the teacher, and from the teacher to the professor, but only one-way communication from the student to the professor. In other words, the professor always communicates his decisions through the teacher to the student.

The professor builds up an "image" or model of the way the student is learning. The model, therefore, is not known in advance of the student's performance on the program, but the kinds of information used to build it are known. Once certain things are determined about the student's behavior, then the strategy or content set can be changed. For example, if latency of response was used initially and the student continued to respond at or above a specified rate over a set of steps, then the professor may decide that use of response latency can be dropped out of later decisions. If student performance is within certain limits of accuracy, the professor may change the tolerance limits used in giving the student knowledge of results.

The professor might use response rate as information about motivation level and this might be derived from a comparison of the present student's rate with the average of other students on the same steps. Thus, the professor may use records of a large number of different students in making a decision about one. Another basis for dynamic decision making might be a predicted error rate as determined by a correlation of aptitude and errors using a standardization sample of students. Either zero order or multiple correlation might be used in computing the expected value and then this would be used in accomplishing the transformation. If this were done, then performance that was better than expected might be transformed into an order from the professor to

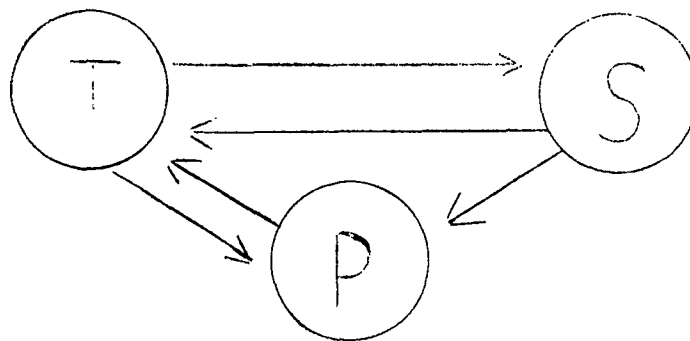


Figure 5. Flow diagram of student-teacher-professor interactions.

the teacher to skip particular sections of the content set or to reduce all sets of review steps scheduled to appear later on, etc. If, on the other hand, the performance of the student were as expected, then the teacher would continue with the program as originally planned. If, however, performance were lower than expected, then the professor might specify a change in strategy which would result in more remediation for each student error.

This description of the tutorial phase of instruction completes the outline of a general model of teaching as accomplished by an adaptive teaching machine system. The process is accomplished in two phases: the pre-tutorial and tutorial decision processes. The pre-tutorial process involves the selection of a program for a particular student on the basis of his past ability and performance records. The tutorial process involves two interacting processes. In the first, the Teacher Function selects an instructional sequence on the basis of immediate student performance. In the second, the Professor Function makes changes in the program if the performance of the student exceeds certain bounds.

To people who are familiar with present-day teaching machines, this description may seem overly fancy. However, the development of complex computer-based systems requires a careful analysis of the teaching machine process, and it is hoped that this model provides a start in that direction.

In a subsequent section, a computer-based system which incorporates some of the ideas developed here is described.

Functions Performed During The Implementation Of A Program

In order to implement a program it is necessary to perform certain functions. Thus, if a machine is to make use of the student's responses in making decisions, it must somehow acquire the responses and put them in usable form for evaluation. If the evaluation is to make use of past response information, then the machine must have memory capabilities for responses as well as for standards of evaluation. If the evaluation is to be used to select content units for presentation, it is necessary to have a library of content units available and to have a means of choosing the way in which the units are presented; and finally, display functions are needed to present the selected content units and to give the student knowledge of results.

Within the broad classification of response acquisition, evaluation, selection, and display functions there are certain operations which must be performed by every teaching machine. While all of these are not always stated explicitly, each is nevertheless involved in every instance of tutorial instruction. Furthermore, depending upon the degree of automation, each of these functions is performed either by a device or a person. Consequently, a description of machine operation is a statement that considers each of these functions.

When the learner accomplishes one or more of these functions, he is, in fact, performing a teacher's function, not a learner's function. In other words, students can teach themselves, but this is just a special case of the general concept of teaching described here. In short, from a systems point of view it is not critical that any function be performed by either a machine or a person. Nor is it critical who the person is if it is not performed by a machine. The nature of the function is the same in any case.

In teaching machine operation, the actual transformations specified by each function determine the teaching strategy. When these are used in conjunction with a particular set of content units, the teaching program is determined. Thus, the output of the pre-tutorial decision process is implemented by specifying the particular transformation accomplished by each function. For example, evaluation can be made in terms of any number of responses and selection can be made from various subsets of the total content set. The remainder of this section is concerned with a description of various machine functions.⁵

Response acquisition. The response function involves the communication of the student's response to the machine and the transformation of it into usable form. The characteristics of the equipment which is used to perform this function depend on the response mode (written, typed, punched, etc.) and the form of the response information needed for purposes of evaluation. In a simple machine which incorporates a linear program, the response is written and stored on paper for later analysis. There is no need to use the response for evaluation purposes since there is no decision making involved.

In the case of computer-based systems the versatility has been greatly increased by using typewriters at the student stations (Bitzer, 1962). The response is then transformed into a form which can be used for decision making. Also, response measures other than the student's response could be used (e.g., latencies), with the appropriate equipment.

Evaluation. Involved in the evaluation function is the comparison of the response measure with a standard. The input to a comparator may be either the same symbol system used in the display or a transformation of these symbols.

⁵These are based on Stolurow's ten critical requirements of a teaching machine. (1961, Chapter II).

Whichever is used must permit a comparison of the student's response with that which is correct--the standard. If a computer does the comparing for the student, as does PLATO (Bitzer, 1962), then the response has to be transformed into computer language so that it can be compared with the answer that is stored in the computer. Often, however, the comparison function is accomplished by the student himself when he looks at his response under glass and compares it with the correct one which is printed in a particular place on the display material (e.g., Skinner's machines and those like it).

While a student can perform this function, a machine such as a computer is required if the student is told he is wrong without also being told the right answer. Furthermore, if response rate is part of the critical response information then the comparison function is performed using response latency as well as correctness and the process is more complicated. By providing the student with time data, he could compare his response latency to the standard, but requiring him to do so, particularly if he is young, might destroy the rhythm and continuity of his response sequence. In the automated teaching of card punching (Pask, 1960, SAKI System), typewriting or stenotyping (Uttal, 1962) both the accuracy and the latency of response are compared to appropriate standards. For these purposes a computer-based system seems indicated.

The output of a comparator must be arranged and stored in a form which permits its use in immediate decision making or subsequent analysis. This is accomplished by the collating and recording function which involves accumulation and arranging of information about the student as the program progresses. Generally it is desired that the individual responses be related to relevant steps of the program. This is useful for program revision and for individual

diagnosis once a program has been developed. For individual diagnostic purposes the accumulated record of a student's responses to selected sets of steps (e.g., all steps relating to a concept) may be used in making dynamic decisions about the teaching program (Professor Function).

Selection. The information stored by the collator-recorder is used to select content units for presentation and to choose a particular pace for the student. The content units are in the library of the machine which is used for storage of material for display. Storage may be in the form of electrical charges, perforated tape, punch cards or printed as in film or on paper. The transformation differs depending upon the form of storage and the amount of memory used. Memory may be in more than one form and place as when a computer stores the frame locations of steps in its core storage and uses these to select information stored on film in a projector.

The two problems associated with this function are those of capacity and access time. Available systems differ in their limitations on storage, e.g., core storage of computers and film capacity of projectors, for PLATO (Bitzer, 1962); the IBM 1410 System developed by Uttal; and SOCRATES developed by Stolurow.

The input to the library is provided by a selector; the output is a particular set of steps that is displayed to the student. In terms of the model, the selector function is equivalent to the teacher in that it uses response dependent information to choose content units for subsequent display.

Part of the selector function is the provision of knowledge of results (KR) in which the student is given feedback relating to his response. The separation of this function from the selection of content units is desirable if there is

likely to be a delay in the display of the step providing new information due to the searchtime involved in locating the step. If the student rather than a machine provides knowledge of results, then the feedback with a linear style of program is just as rapid for wrong as right responses. In a branch style of program the times can be manipulated so that they are not necessarily the same. If a machine such as a computer is used to provide KR feedback then the times can be equated or not depending upon their relative effectiveness.

Display. The display function represents a transformation of a designated content sample into a usable set of symbols with certain constraints. The symbols determine the modality (e.g., printed words-visual). The constraints imposed are those pertaining to the form of the display as a set of instructional steps and to duration of its exposure to the student. To illustrate the transformation, the program to be taught might be stored in the system as a perforated tape, but the display used by the student might be printed words on a television screen. To convert the tape perforations into black and white dots on a television screen requires a fairly elaborate transformation process. The engineering details are much more complex than those required to move a piece of paper as in a device like the Konzept-O-Graph and Min/Max (Finn and Perrin, 1962). In the latter two devices, the content is stored in the same form that it is seen by the student. Therefore, the transformation accomplished by display is trivial. The function cannot be ignored, however, for it obviously is significant in the case of a computer-based teaching system.

Part of the total display is knowledge of results. As discussed, the transformation that is required to display this information to the student

may be independent of that involved in the display of content, but it is more common to achieve both by the same transformation, e.g., a new display may appear only if the response to the last one is correct or knowledge of results may be given to all students when they go to the next frame as in a linear program. Obviously, a general model has to allow for the independence of these two transformations as well as for their complete dependence upon one another.

While all the above functions must be included in any teaching machine, examples were given which illustrated that the manner in which the functions are carried out varies widely. In the simplest machines, they are performed by the learner himself. In more complex machines, many are performed automatically, and it is expected that in computer-based systems all functions will be performed by the computer and its peripheral equipment.

While a number of computer-based systems have been developed and are being used experimentally at the present time, their use in automated instruction is often less than it might be. Too frequently they simply ape the simpler systems. There is little point to using a computer to simulate a Skinner type of machine, a programmed text, or even a "Tutor Text" (Crowder, 1960).

Pask (1960), on the otherhand, has developed machines that use computers to do things which it would be impossible or impractical to accomplish by any other means. A number of systems have been developed with the potentiality for use in some of the ways described.

It is quite apparent from the brief description given of the functions involved in tutorial instruction that there are many different ways in which

each function can be accomplished. The specification of a set of functions for eventual use in machine design has to be supplemented with more specific descriptions to achieve a workable system. Table 1 is a very brief description of some of the major features of several computer-based systems currently being used.

The descriptions do not reveal all that needs to be known in evaluating a system of this type for a particular purpose. It would be necessary to consult the references and the personnel involved in the system to do this.

Description of SOCRATES System

SOCRATES is a newly designed system which has not been described in the literature before. It is being developed as an adaptive teaching machine system along the lines of the model which was presented in earlier sections. It will be described here in order to illustrate the features of a modular system designed to conduct educational research of the kind described in the next section. The system gets its name from its description--System for Organizing Content to Review And Teach Educational Subjects.

As in the case with even the simplest machines, the basic plan is a student, a machine, and a program in interaction as represented in Figure 6.

The display unit for SOCRATES is a rear view projection screen on which a filmed image is projected for the student to see. His response will be the selection of one of ten buttons. This will be fed into a computer. The computer will have stored in its memory instructions that will tell it what to do

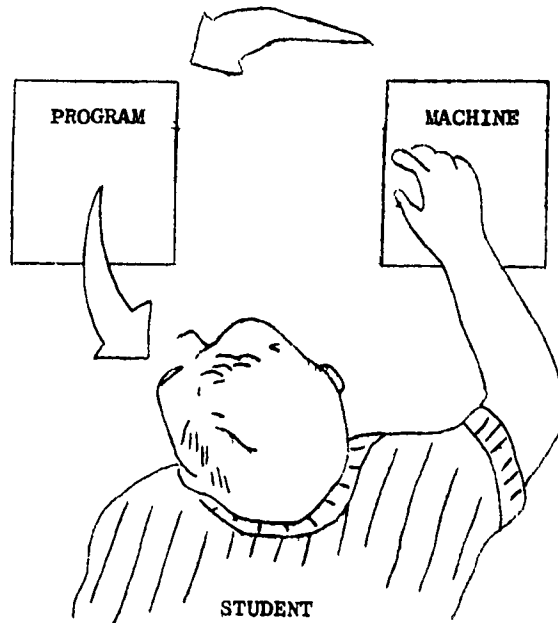
Table 1

Some Current Computer-Based Systems

	(Uttal) ¹	PLATO ²	SOCRATES	CLASS ³	(Licklider) ⁴
Number of Input Stations	8	2	14	20	2
Input Equipment	Stenotype; IBM 838 I/O Typewriter	Special Keyboard	Autotuter; Standard Typewriter	Special Keyboard	Typewriter
Library (media)	Slides	Slides	35 mm. film strip	35 mm. film strip	?
Computer	IBM 1410	CDC 1604	IBM 1620	Philco 2000	PDP-1
Programing Systems	Fortran, Autocoder	Codap	Fortran, SPS	?	?
Display	Typewritten text; projected image	Projected slide image on TV screen	Projected film strip image	Typewritten text; pro- jected slide image	Typewritten text

¹See Uttal, 1962.²See Bitzer, et al on Coulson (Ed.) 1962. Pp. 205-216.³See Coulson (1962), Pp. 191-205.⁴See Licklider in Coulson (Ed.) 1962. Pp. 217-239.

Figure 6



with this information. One thing it will do is make a permanent record of the response in such a way that it can be collated with the display that was shown at the time. Another thing it will do immediately is to tell the student that he is right or wrong through a separate KR circuit which will turn on a green or red light in the display. The computer also will scan the stored library key and select from it a step of a program to be displayed. The library is a film containing a large number of frames on each of which can be stored anything that can be put on a page of a book.

The flow diagram of SOCRATES is shown in Figure 7. The response acquisition and display functions are carried at the student station, and the evaluation and selection functions are carried out by the professor and teacher. While the Teacher Function is involved with the implementation of the current program the professor searches for new programs if the present one is not effective. The administration controls the entire process and a large capacity remote computer may be used in cases where the on-line computer cannot handle the operations. Also, when a student leaves a station, a command is relayed to the administration (via the "recess" path) and action is taken to record the current status of the system (e.g., last frame presented, topic, strategy, content set) and the student's response record. When the student returns, this information is fed into the system and the status is restored.

The actual equipment system is diagramed in the next figure. As indicated the student stations, of which there are 14, can be different and include Autotutors as well as the PerceptoScope teaching machine, and IDIOT (The Illinois Device for Independent Operation and Teaching). A typewriter also could be used.

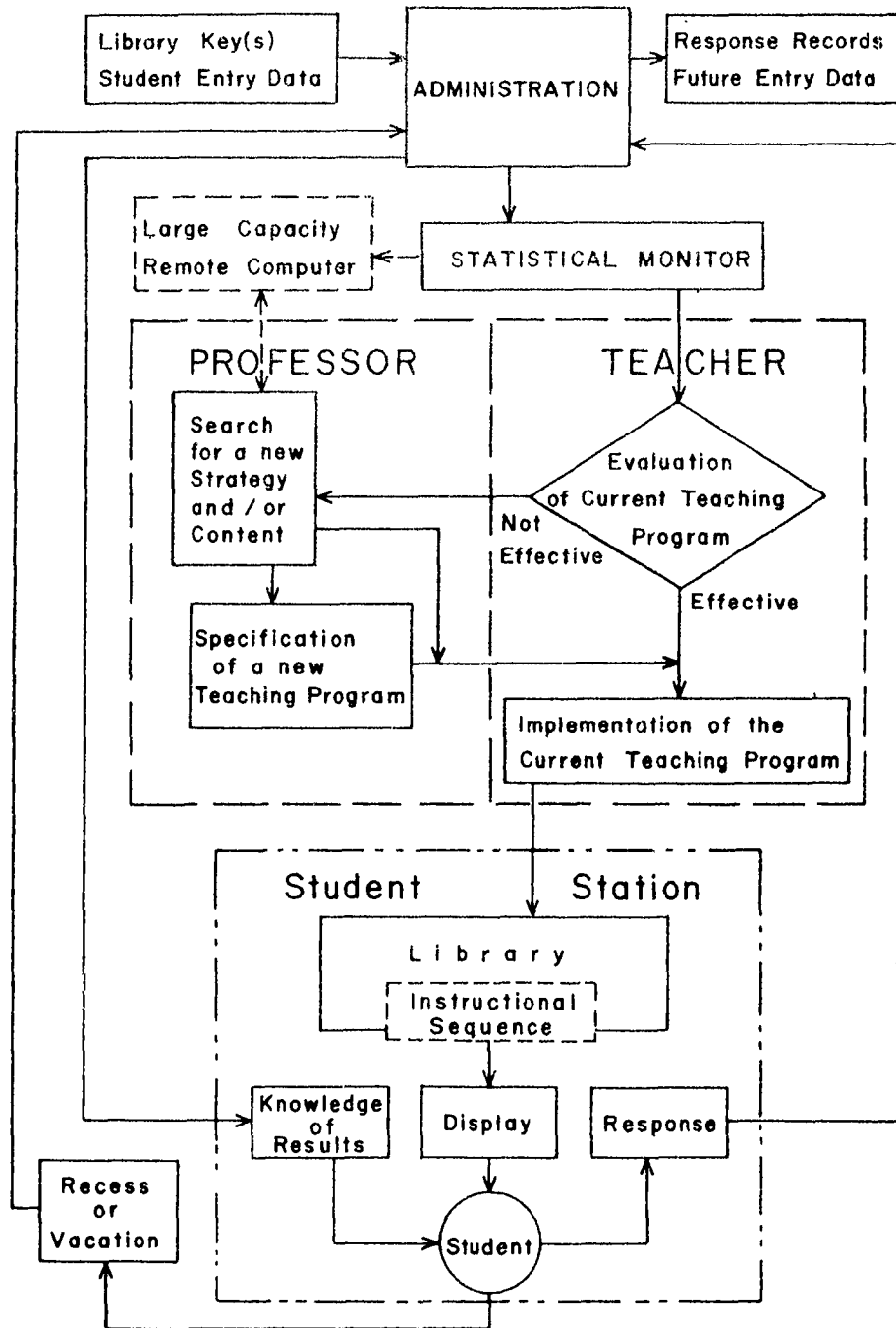


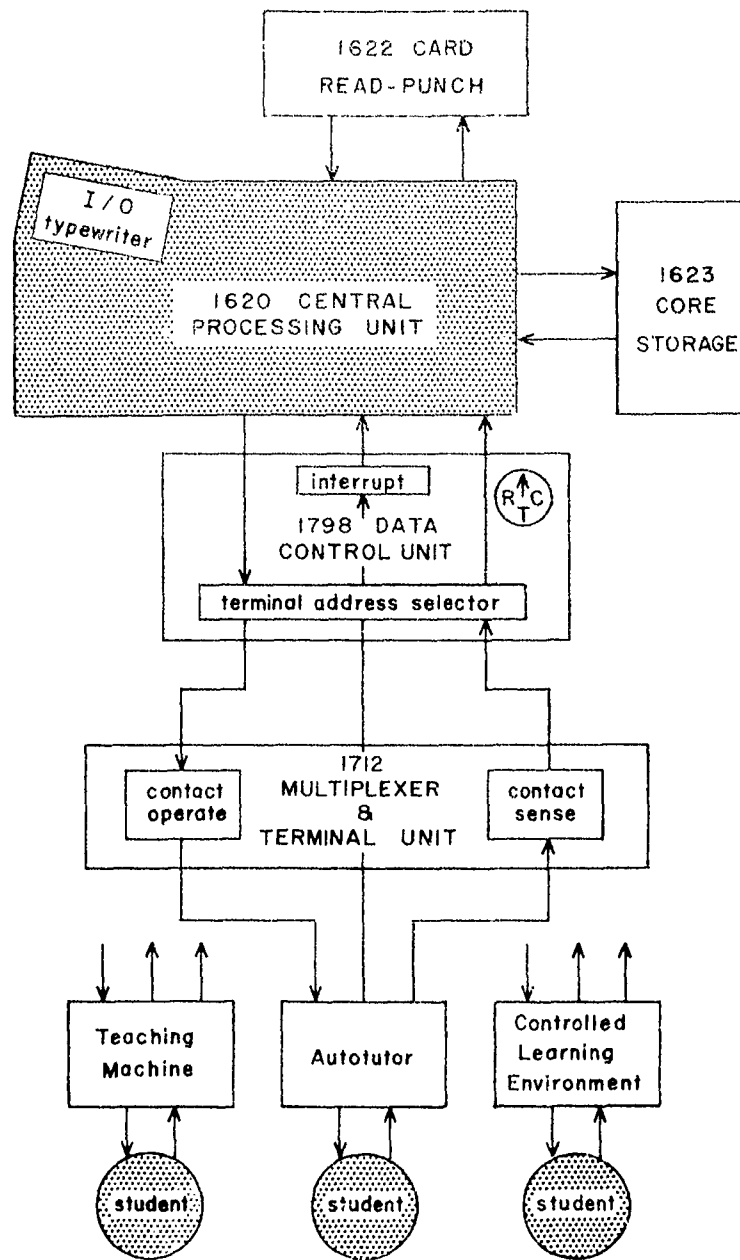
Figure 7. Flow Diagram of Socrates System

Each station will have its location code or address. It will be wired directly to the computer. The professor rules and associated memory blocks will be stored in the computer. The teacher will use the rules which were specified in the pre-tutorial phase. The associated content set will be chosen from the library. Not all students need to start at the same place.

Each time a student makes a response, an interrupt signal is generated. The administration interrupts its activity (which may be just waiting for a student to respond) to identify which student made the response and what response it was. Multiple responses are queued and accepted in order. The administration stores the student's response and determines the time and the latency of the response (the time that the step was displayed). Pacer requirements are checked to see that the response should be accepted and what display should be made. The response is checked against the library key to determine its correctness. The student is signalled knowledge of results of this response. Considering the student's latency history, his response history, and his characteristics, the teacher determines the instructional step to be displayed from the library key. The decision results in a signal which transfers the instructional step from the library to the display unit. The administration then creates a permanent record of the student's response latency and instructional step which can be used by the professor for subsequent decision making.

Applications of Computer-Based Teaching Machine Systems

The most apparent application of computer-based teaching machine systems is to study the relative effectiveness of particular teaching programs. In



The SOCRATES System

Figure 8 Equipment Interconnections

spite of all the research on teaching, we do not have a very firm grasp of the critical variables in instruction. The determination of the relative effectiveness of the variables in teaching is a basic and high priority topic. The results of this research should contribute materially to a theory of teaching.

Equally important is the research on the relationship between aptitude (individual differences) and programs. This is the problem previously identified as the pre-tutorial problem since it is this decision sequence which must select a program to use in teaching a student. An important task is to select appropriate ways of specifying useful levels of individual differences among the Ss. One way to determine the utility of a set of distinctions is to use a single program with presumably different Ss to see if different outcomes result. There should be at least as many different outcomes as there are meaningful distinctions among the Ss. The results of these studies also would contribute to a theory of teaching.

Basic research on learning is a third area of study to which the teaching machine systems can be applied, e.g., paired associates learning (Licklider, 1961). In particular, a computer-based teaching machine system should provide a means for studying more complex designs to determine the interactions of a higher order that might be important in developing strategies. Furthermore, such a system would permit the more rapid conduct of these studies, reducing the time lags that now exist in basic research between hypothesis formation and its test.

It would seem that the real future of programmed instruction lies in the development of knowledge about the more complex strategy of instruction which

cannot be implemented in any other way than by the use of the computer-based system. Not only the teaching strategies, but also the professor variables, need to be studied. Some specific variables that seem promising at this time both in terms of educational implications and practicality were those previously mentioned.

Two similar and interrelated instructional problems seem to be undifferentiated in current thinking about the requirements for automated instruction. One of these, and probably the more prominent of the two, is the pre-programmed self-instruction (PSI) which has enjoyed widespread attention in recent years. In developing this form of instruction, a strategy is developed through pragmatic research and, once it is built into a set of instructional materials, they are used uniformly for all students to whom the instruction is given.

A second form of instruction has gone almost unnoticed. It can be described as self-programmed individualized education (SPIE). In this form of instruction, the learner is presumed to be mature and knowledgeable in that he has acquired substantial amounts of information relating to one general area but has gaps in what he has already mastered. In other words, with SPIE, the learner has some knowledge and wants to complete his repertory of information. He is presumed to be capable of defining for himself the gaps that he has in his knowledge and of making useful decisions about the sequence he wants to follow to fill in these gaps. For this latter reason, the instruction is called self-programmed. Under these conditions, the strategic problem of instruction can be turned over to the student once the student has been provided with general information about programing. Thus, the system used must be developed to provide appropriate and critical terms for the topical areas to be

selected for study by the student. These would provide a kind of inventory of the available steps. The student's task would be to select appropriate ones of these in a particular order that meets his own needs as he perceives them at that point in time.

Finally, a computer-based teaching facility could provide data that would soon make it obsolete. The research on the problems mentioned will also provide data relating to the redesign of the system used and these very data are what will permit even more substantial research at a later time.

Implications of the Model for Research

The question is often asked about the use of machine vs. the use of a book format. This seems to be a relevant question on the surface but turns out to be specious on analysis for two reasons. First, it is specious for the same reason that a comparison between automated instruction and live instruction is invalid. Any study that compares a program with a teacher is dealing in samples of a single case. Furthermore, each sample (the program and the teacher or the program without a machine and the program with it) is of unknown merit relative to other possible exemplars of its universe. Thus, a subsequent study could find opposite results.

Second, the question is wrong. The one that needs to be asked is "How simple or how complex a machine should one use?" By simple and complex we mean the number of different teaching programs the machine can make available for use. The machine vs. book question cuts things differently from the question of having one vs. having many programs available for use. Consequently, the pertinent variable does not get manipulated when studies are conducted in which a group using a machine is compared with one using a book. The results could

come out differently in different experiments depending upon the relative number of programs used by the machine compared to the number used by the book in coping with different levels of student entry behavior. The typical finding from the studies reported thus far (Goldstein and Gotkin, 1962) is that there is no difference in the effectiveness of a machine and a book. This result is to be expected since in these studies the number of teaching programs was equal for the two. As long as the number of different groups of Ss classified in terms of entry behavior was the same for those using the book as for those using the machine, the results should be the same in terms of the number of different classes of outcomes.

The implication, then, is that research should be conducted on the relationships between input behaviors (aptitude and performance), programs (strategies and content) and outcomes (time, final performance level, and topic covered). As a result, questions such as the following would be answered: For a given subject area and objective, how many strategies and content sets should the system have in order to teach effectively with a given input range? What are the characteristics of strategies and content units which work best for particular aptitudes? What are the natures of the limits which aptitude places on the outcomes which are achievable?

During the course of answering these questions tables such as those in Figures 2, 3, and 4 would be filled in, and a true technology of teaching would result. Computer-based systems would be used to find the best program for each student and to implement and monitor tutorial instruction.

Is a Computer-Based Teaching Machine System Practical?

If we mean by this question "Is a computer-based system something which only the most wealthy can afford?" then the answer is no. In fact, it may be

just the reverse (Bitzer, NEC Proceedings, 1962), for it has been estimated that with the PLATO system as many as 1,000 students could be taught up to eight different programs simultaneously and the computer would be working only about one-third the time. Therefore, the computer could be used for other things, e.g., computing scores for students or step difficulty values..

In terms of cost per student hour of instruction it is difficult to get a firm estimate; however, one recently computed by Bitzer (1962) suggests that the cost based upon the use of existing hardware could be brought down to ten cents per hour per student. This indicates that the computer-based teaching machine system is not only practical for research, but will subsequently be so for regular instruction as well.

Stolurrow

References

- Ashby, W. R. An introduction to cybernetics. London: Chapman and Hall, 1962.
- Bitzer, D. L., Braunfeld, P. G., and Lichtenberger, W. W. Plato II: A multiple student, computer controlled, automatic teaching device. In J. E. Coulson (Ed.), Programmed learning and computer based instruction. New York: Wiley, 1962, Pp. 205-216.
- Coulson, J. E. (Ed.) Programmed learning and computer-based instruction. New York: Wiley, 1962.
- Coulson, J. E. A computer-based laboratory for research and development in education. In J. E. Coulson (Ed.), Programmed learning and computer-based instruction. New York: Wiley, 1962. Pp. 191-204.
- Crowder, N. A. Automatic tutoring by intrinsic programming. In A. A. Lumsdaine and R. Glaser (Eds.), Teaching machines and programmed learning. Washington, D. C.: National Education Association, 1960.
- Crowder, N. A. The arithmetic of computers. New York: Doubleday, 1960.
- English, H. B. How psychology can facilitate military training: A concrete example. J. appl. Psychol., 1942, 26, 3-7.
- Finn, J. D., and Perrin, D. G. Teaching machines and programed learning 1962: A survey of the industry. Occasional Paper No. 3, Tech. Develp. Proj., 1962.
- Goldstein, L. S., and Gotkin, L. G. A review of research: teaching machines vs. programed textbooks as presentation modes. J. Programed Instr., 1962, 1, 29-35.
- Huskey, H. D. Automatic computers and teaching machines. In J. E. Coulson (Ed.), Programmed learning and computer-based instruction. New York: Wiley, 1962.
- Jevons, W. S. The principles of science. New York: Dover, 1958.
- Licklider, J. C. R. Preliminary experiments in computer-aided teaching. In J. E. Coulson (Ed.), Programmed learning and computer-based instruction, 1961. Pp. 217-239.

Stolurow

Lumsdaine, A. A., and Glaser, R. (Eds.). Teaching machines and programmed learning. Washington, D. C. National Education Association, 1960.

Mellan, I. Teaching and educational inventions. J. Exp. Educ., 1936, 4, 291-300.

Ordahl, Louise E., and Ordahl, G. Qualitative differences between levels of intelligence in feeble-minded children. J. Psycho-asthenics, Monogr. Supplement 1915, 1, 3-59.

Pask, Gordon. Adaptive teaching with adaptive machines. In A. A. Lumsdaine and R. Glaser (Eds.), Teaching machines and programmed learning. Washington, D. C.: National Education Association, 1960. Pp. 349-366.

Pask, Gordon. Electronic keyboard teaching machines. In A. A. Lumsdaine and R. Glaser (Eds.), Teaching machines and programmed learning. Washington, D. C.: National Education Association, 1960. Pp. 336-348.

Pressey, S. L. A simple apparatus which gives tests and scores--and teaches. Sch. and Soc., 1926, 23, 373-376.

Pressey, S. L. Further attempts to develop a mechanical teacher. Amer. Psychologist, 1950, p. 1.

Pressey, S. L. Develop. and appraisal of devices providing immediate automatic scoring of objective tests and concomitant self-instruction. J. Psychol., 1950, 29, 417-447.

Rigney, J. W. Potential uses of computers as teaching machines. In J. E. Coulson (Ed.), Programmed learning and computer-based instruction. New York: Wiley, 1962. Pp. 155-170.

Senders, J. Adaptive teaching machines. In J. E. Coulson (Ed.), Programmed learning and computer-based instruction. New York: Wiley, 1962. Pp. 129-133.

Skinner, B. F. Science of learning and the art of teaching. Harvard Educ. Rev., 1954, 24, 86-97.

Skinner, B. F. Teaching machines. Sci., 1958, 128, 969-977.

Stake, R. E. The teaching machine: Tool of the future or passing fancy. Phi Delta Kappan, 1963, XLIV (6), 247-249.

Stolurrow

Stolurrow, L. M. Teaching by machine. Coop. Res. Monogr. No. 6
Washington, D. C., U. S. Office of Education, 1961.

Teager, H. M. Systems considerations in real-time computer
usage. In J. E. Coulson (Ed.), Programmed learning and
computer-based instruction. New York: Wiley, 1962.

Thorndike, E. L. Educ. New York: Macmillan, 1912. Pp. 164-167.

Uttal, W. R. My teacher has three arms! IBM Res. Rep. RC-788,
September 15, 1962.

Uttal, W. R. On conversational interaction. In J. E. Coulson
(Ed.), Programmed learning and computer-based instruction.
New York: Wiley, 1962. Pp. 171-190.

Uttal, W. F., Charap, Marilyn, and Maher, Anna. The computer
tutoring of stenotypy: A preliminary report. IBM Res.
Rep. RC-663, April 11, 1962.

Valentine, J. A. The development and use of phase checks. In
N. Hobbs (Ed.), Psychological research on flexible gunnery
training. Washington: Govt. Printing Off., 1947.
Pp. 119-139.

Vallance, T. R., and Shrader, W. B. The evaluation of training
devices and procedures. In N. Hobbs (Ed.), Psychological
research on floxible gunnery training. Washington: Govt.
Printing Off. Pp. 141-258.

Wiener, N. Cybernetics. New York: Wiley, 1948.